BDNR Course

BDNR · Non-Relational Databases
M.EIC · Master in Informatics Engineering and Computation

Sérgio Nunes
Dept. Informatics Engineering
FEUP · U.Porto

Based on Chapter 1 from NoSQL Distilled, Sadalage and Fowler (2012)

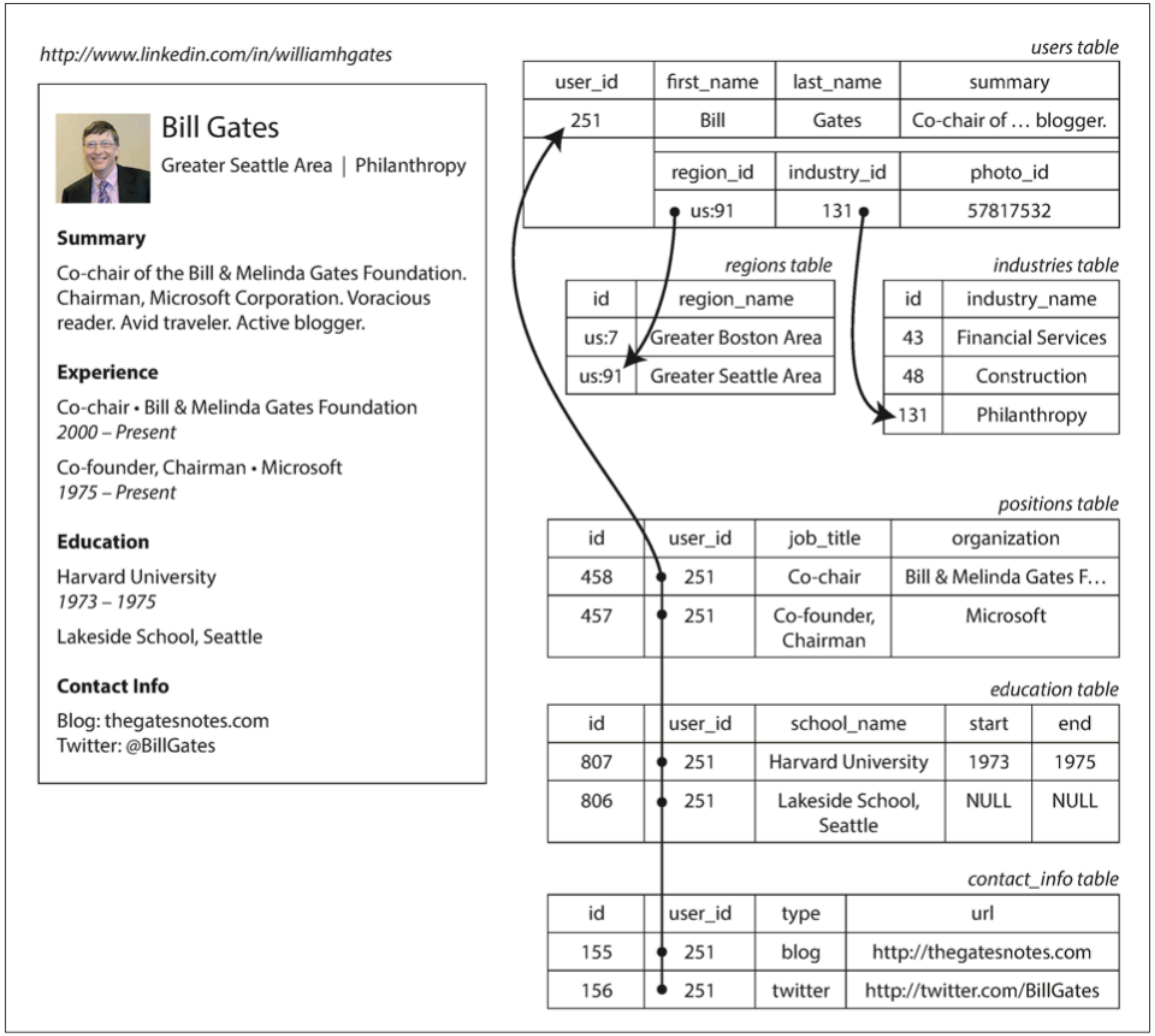
Today's Plan

- Context and motivation
- Course presentation
  - Topics
  - Evaluation
  - Students' Projects

Context and Motivation

Relational Databases

- Relational databases are the default choice for data storage, a *de facto* standard.
- Recall the value of relational databases:
  - Persistence — keeping large amounts of data store for quick and easy access;
  - Concurrency — allow and control simultaneous accesses to the same data through transactions;
  - Integration — share a single data store that allows for integration at application level;
  - Standard — an adopted standard for modeling and manipulating data that shares the same core concepts despite differences in implementation.



users table

user_id	first_name	last_name	summary
251	Bill	Gates	Co-chair of … blogger.

region_id	industry_id	photo_id
us:91	131	57817532

regions table

id	region_name
us:7	Greater Boston Area
us:91	Greater Seattle Area

industries table

id	industry_name
43	Financial Services
48	Construction
131	Philanthropy

positions table

id	user_id	job_title	organization
458	251	Co-chair	Bill & Melinda Gates F…
457	251	Co-founder, Chairman	Microsoft

education table

id	user_id	school_name	start	end
807	251	Harvard University	1973	1975
806	251	Lakeside School, Seattle	NULL	NULL

contact_info table

id	user_id	type	url
155	251	blog	http://thegatesnotes.com
156	251	twitter	http://twitter.com/BillGates

Figure 2-1. Representing a LinkedIn profile using a relational schema. Photo of Bill Gates courtesy of Wikimedia Commons, Ricardo Stuckert, Agência Brasil.

Image from Designing Data-Intensive Applications, Kleppmann (2017)

Why NoSQL?

Impedance Mismatch

- "Impedance mismatch", is an expression from electronics referring to the difference between the electrical resistance of two circuits, resulting in inefficient power transfer.
- In the context of information systems, refers to the difference between distinct data representations, most notably between the relational model and in-memory data structures.
- In the relational model, data is organized and manipulated using relations, i.e. sets of tuples (name-value pair), which only allow for simple structures, e.g. they cannot contain nested elements.
- This limitation is not true for in-memory data structures, e.g. arrays, lists).
- As a result, to use richer data structures a translation is necessary between to and from a relational representation.

Impedance Mismatch

➤ Impedance mismatch exists when two different representations require translation.

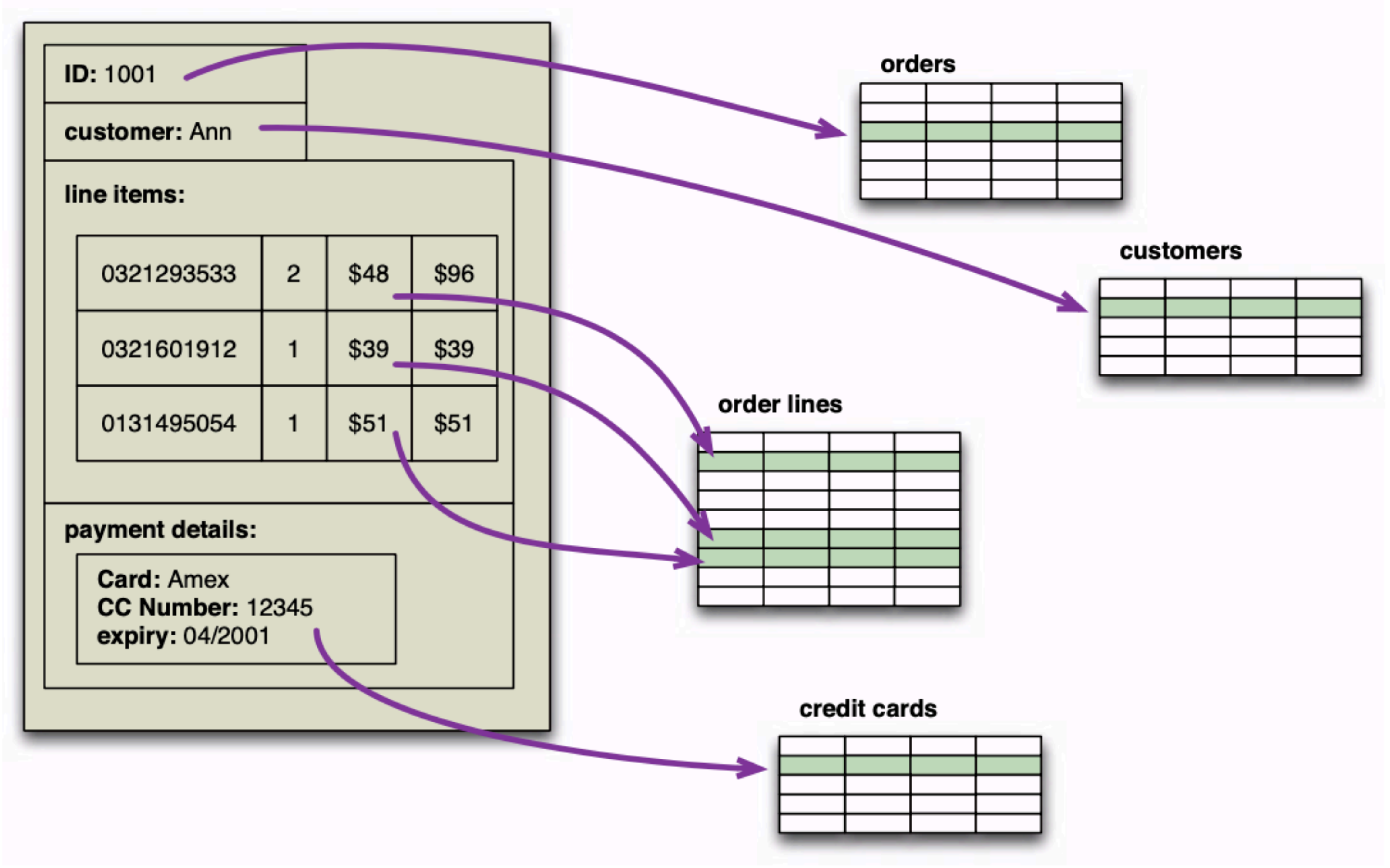


Figure 1.1 *An order, which looks like a single aggregate structure in the UI, is split into many rows from many tables in a relational database*

Image from NoSQL Distilled, Sadalage and Fowler (2012)

Impedance Mismatch

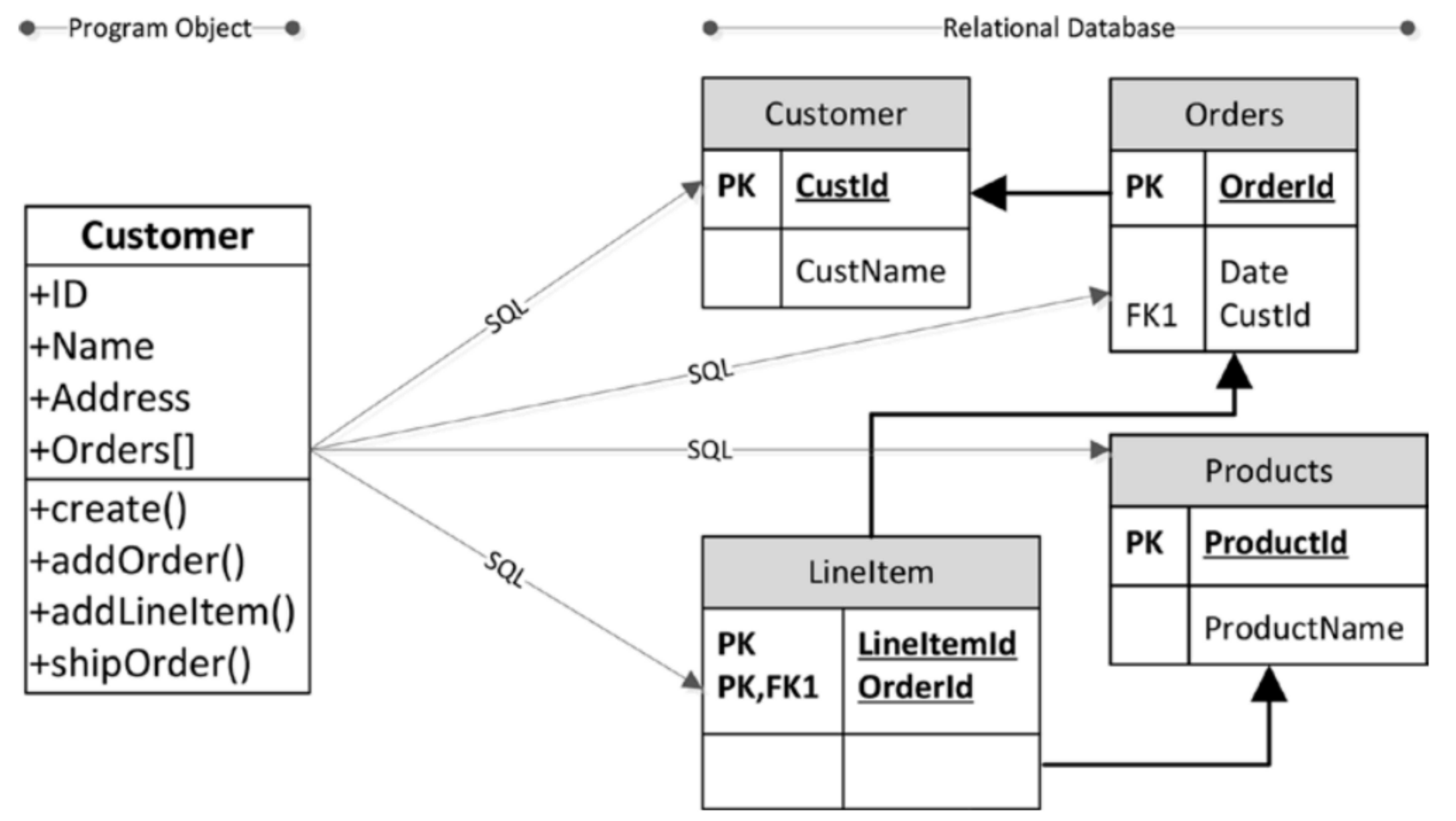


Figure 1-5. *Storing an object in an RDBMS requires multiple SQL operations*

Image from Next Generation Databases, Harrison (2016)

Big Data

- Since the 2000s, we witness to an era of consolidation in web platforms (big tech).
- Results in network effects, i.e. big platforms are more attractive to users.
- This large scale impacts many dimensions — data collection, heterogeneity, storage.
- To handle this growth, computational infrastructures can scale up (i.e. bigger machines to handle more load) or scale out (i.e. more machines to spread the load).
- As infrastructures moved towards clusters, relational database solutions revealed scaling problems adapting as they rely on shared disk subsystems. Solutions exist to overcome these problems (e.g. sharding) but they are "unnatural".
- This mismatch between relational databases and clusters let to the development of alternative routes for data storage solutions.

NoSQL Solutions

- Google and Amazon have greatly influenced development in this area, leading early large-scale implementations of data storage solutions based on distributed low-cost components.
- Google BigTable [1] and Amazon Dynamo [2] landmark papers inspired projects and experiments to develop alternative data storage solutions.
- The name "NoSQL" was a convenient handle to group a diverse set of technologies, although a clear and coherent definition still lacks.
- The central characteristics commonly found in NoSQL solutions are: do not use the non-relational paradigm, have a relaxed consistency model, and adopt a schemaless data model.

[1] Bigtable: A Distributed Storage System for Structured Data
[2] Dynamo: Amazon’s Highly Available Key-value Store

Polyglot Persistence

- Polyglot (adjective), knowing or using several languages.
- Polyglot persistence, using different data stores in different circumstances depending on the nature of the data and how we want to manipulate it.

- Relational databases are one option for data storage (not "the" option).
- Other options exists and typically coexist in an organizational data infrastructure.

Summary

- Relational databases were, and still are, very successful technologies to provide data persistence, concurrency control, and an integration mechanism.
- Limitations of the relational model include the object-relational mismatch and the difficulty in scaling up or out.
- "NoSQL is an accidental neologism (i.e. new word)", whose characteristics include: non-relational, weaker consistency models (scale out is easier), and schemaless data models.
- One of the most important results is polyglot persistence, i.e. diverse data storage ecosystems — one size does not fit all.

Course Presentation

BDNR

- First edition of BDNR.
- M.EIC 1st-year elective course from the area of Information Systems.

- Classes will be on Tuesdays, 14h00 (3h), at FEUP (B203).
  - Topic presentation and discussion;
  - Laboratorial activities;
  - Project development and discussion;

Course Objectives

- The BDNR curricular unit aims to prepare students to know, understand, design and develop solutions based on non-relational database paradigms and technologies to support information systems.

- Specific objectives:
  - Know and understand the main concepts and paradigms of non-relational databases;
  - Enable students to analyze, design, implement and evaluate non-relational databases;
  - Design the storage and interrogation component of systems based on non-relational models.

Learning Outcomes

- Recognize the situations in which relational databases are not the adequate solution for the storage and interrogation of data.
- Identify and describe the different models of non-relational databases and the typical situations of use of each one of them.
- Design, implement and interrogate databases built according to different non-relational approaches.
- Analyze the challenges associated with complex large-scale scenarios (big data), propose solutions based on non-relational models and understand the limits of each one of them.
- Combine relational and non-relational models in information systems.

Prior Knowledge

- Programming
  - knowledge and practice with programming languages for application development.
- Databases
  - knowledge and practice of data modeling in UML;
  - relational model;
  - SQL language.

Course Syllabus

- Non-relational databases:
  - Introduction and motivation;
  - Current data challenges: size, variability, different paradigms;
  - ACID properties and limits of relational databases;
  - Historical perspective of database management systems.
- Properties of non-relational databases:
  - The CAP theorem and design choices;
  - BASE properties;
  - Consistency and distribution techniques;
  - Joint treatment.
- Paradigms:
  - Key-value databases;
  - Column-based;
  - Document databases;
  - Graph databases;
- Hybrid model databases.

Technologies

- Git for version control
- Docker for labs and projects
- NoSQL technologies
  - Redis
  - MongoDB
  - Cassandra
  - Neo4j

Evaluation

- Distributed evaluation with final exam
- Exam: 40% (multiple-choice with open questions)
- Distributed evaluation: 60%
  - Group project (2 ~ 3 students)
  - Select and explore a NoSQL technology
  - Implement and test the system
  - Report and presentation
- Minimum grade of 40% in each component (required but not enough!)
- Herero-evaluation, plus teacher assessment, may result in different grades for each project member.

Class	Date	Topic (Tuesday 14h00, B203, 3h)	Lab / Project
1	8 Mar	Course Presentation; Introduction to NoSQL.	—
2	17 Mar	NoSQL	Groups + Tech Overview
3	24 Mar	Key-Value Databases (1)	Redis Lab / Project Topic Discussion
4	31 Mar	Key-Value Databases (2)	Redis Lab / Project Topic Selection
5	7 Apr	Document Databases (1)	MongoDB Lab
	14 Apr	*//Easter Holiday//*	
6	21 Apr	Document Databases (2)	MongoDB Lab
7	28 Apr	Column-Oriented Databases (1)	Cassandra Lab
	5 May	*//Queima das Fitas//*	
8	12 May	Column-Oriented Databases (2)	Cassandra Lab
9	19 May	Graph Databases (1)	Neo4j Lab
10	26 May	Graph Databases (2)	Neo4j Lab / Project Reports Submission
11	2 Jun	**Student Projects (1)**	
12	9 Jun	**Student Projects (2)**	

Main Bibliography

- Dan Sullivan
  NoSQL For Mere Mortals
  Addison-Wesley, 2015

- Pramod J. Sadalage, Martin Fowler
  NoSQL Distilled: A Brief Guide to the Emerging World of Polyglot Persistence
  Pearson Education, 2013

- Luc Perkins with Eric Redmond, Jim R. Wilson
  Seven databases in seven weeks: a guide to modern databases and the NoSQL movement
  Pragmatic Bookshelf, 2018

Group Project

Group Project

- Study and explore a NoSQL technology
- Developed in groups of 2 ~ 3 students
- Roadmap (tentative)
  - Setup groups
  - Select a NoSQL technology (e.g. https://db-engines.com/en/ranking)
  - Study the technology and implement use cases
  - Write report on the findings
  - Present in the last two classes

383 systems in ranking, February 2022

Rank Feb 2022	Rank Jan 2022	Rank Feb 2021	DBMS	Database Model	Score Feb 2022	Score Jan 2022	Score Feb 2021
1.	1.	1.	Oracle	Relational, Multi-model	1256.83	-10.05	-59.84
2.	2.	2.	MySQL	Relational, Multi-model	1214.68	+8.63	-28.69
3.	3.	3.	Microsoft SQL Server	Relational, Multi-model	949.05	+4.24	-73.88
4.	4.	4.	PostgreSQL	Relational, Multi-model	609.38	+2.83	+58.42
5.	5.	5.	MongoDB	Document, Multi-model	488.64	+0.07	+29.69
6.	6.	↑7.	Redis	Key-value, Multi-model	175.80	-2.18	+23.23
7.	7.	↓6.	IBM Db2	Relational, Multi-model	162.88	-1.32	+5.26
8.	8.	8.	Elasticsearch	Search engine, Multi-model	162.29	+1.54	+11.29
9.	9.	↑11.	Microsoft Access	Relational	131.26	+2.31	+17.09
10.	10.	↓9.	SQLite	Relational	128.37	+0.94	+5.20
11.	11.	↓10.	Cassandra	Wide column	123.98	+0.43	+9.36
12.	12.	12.	MariaDB	Relational, Multi-model	107.11	+0.69	+13.22
13.	13.	13.	Splunk	Search engine	90.82	+0.37	+2.28
14.	14.	↑15.	Microsoft Azure SQL Database	Relational, Multi-model	84.95	-1.37	+13.67
15.	↑17.	↑35.	Snowflake	Relational	83.18	+6.36	+64.96
16.	↓15.	↓14.	Hive	Relational	81.88	-1.57	+9.56
17.	↓16.	17.	Amazon DynamoDB	Multi-model	80.36	+0.50	+11.21
18.	18.	↓16.	Teradata	Relational, Multi-model	68.57	-0.56	-2.33
19.	19.	↑20.	Solr	Search engine, Multi-model	58.53	+0.00	+7.84
20.	20.	↓19.	Neo4j	Graph	58.25	+0.21	+6.08
21.	21.	21.	SAP HANA	Relational, Multi-model	56.31	-0.61	+6.09
22.	22.	22.	FileMaker	Relational	54.14	-1.72	+7.94
23.	23.	↓18.	SAP Adaptive Server	Relational, Multi-model	49.54	-1.52	-2.71
24.	24.	24.	Google BigQuery	Relational	45.10	-0.52	+9.21
25.	25.	↓23.	HBase	Wide column	43.62	-0.37	-0.87
26.	26.	↓25.	Microsoft Azure Cosmos DB	Multi-model	39.95	-0.09	+8.29
27.	27.		PostGIS	Spatial DBMS, Multi-model	31.02	-0.85	
28.	↑29.	↓26.	Couchbase	Document, Multi-model	30.07	+1.21	-0.59
29.	↓28.	↓27.	InfluxDB	Time Series, Multi-model	29.34	-0.74	+3.09
30.	30.	↓29.	Firebird	Relational	26.36	-0.91	+3.02
31.	↑32.	↓28.	Memcached	Key-value	25.77	+0.43	+0.10
32.	↓31.	↓31.	Amazon Redshift	Relational	25.41	-0.44	+3.69
33.	↑34.	↑34.	Spark SQL	Relational	23.28	+0.34	+4.83
34.	↓33.	↓30.	Informix	Relational, Multi-model	22.31	-0.63	-0.03
35.	↑38.	↑42.	Microsoft Azure Synapse Analytics	Relational	19.80	+1.31	+9.44
36.	↑37.	↓33.	Netezza	Relational	19.51	+0.24	+0.86
37.	↓35.	↓32.	Vertica	Relational, Multi-model	19.29	-0.61	-1.27
38.	↓36.	↓37.	Firebase Realtime Database	Document	19.15	-0.21	+3.15
39.	39.	↓36.	Impala	Relational, Multi-model	18.91	+0.45	+2.84
40.	40.	↓38.	CouchDB	Document, Multi-model	17.45	+0.81	+1.84
41.	41.	↓39.	dBASE	Relational	14.70	+0.32	+1.41
42.	↑43.	↓40.	Presto	Relational	14.28	+0.91	+1.96
43.	↓42.	↓41.	Greenplum	Relational, Multi-model	13.52	-0.44	+1.31
44.	44.	↑54.	ClickHouse	Relational, Multi-model	12.82	+0.41	+5.16
45.	45.	↑46.	Amazon Aurora	Relational, Multi-model	12.23	+0.49	+3.26
46.	46.	↑48.	etcd	Key-value	11.71	+0.16	+3.07
47.	↑48.	47.	Datastax Enterprise	Wide column, Multi-model	10.75	+0.86	+1.95
48.	↓47.	↓44.	H2	Relational, Multi-model	10.11	+0.12	+0.80
49.	↑50.	50.	Hazelcast	Key-value, Multi-model	9.50	-0.03	+1.08
50.	↑51.	↓43.	MarkLogic	Multi-model	9.46	+0.27	+0.00
51.	↓49.	↓45.	Realm	Document	9.42	-0.17	+0.27
52.	↑53.	52.	Kdb+	Time Series, Multi-model	9.11	+0.34	+1.33
53.	↓52.	↓51.	Google Cloud Firestore	Document	9.06	+0.14	+1.01
54.	54.	↓53.	Algolia	Search engine	8.92	+0.55	+1.18
55.	↑57.	↓49.	Oracle Essbase	Relational	8.41	+0.99	-0.20
56.	56.	↑60.	Microsoft Azure Search	Search engine	7.90	+0.38	+1.63
57.	↓55.	57.	Sphinx	Search engine	7.63	-0.40	+1.07
58.	58.	↓56.	CockroachDB	Relational	7.47	+0.50	+0.87
59.	59.	↑65.	SingleStore	Relational, Multi-model	7.33	+0.67	+2.12
60.	↑64.	↑72.	Riak KV	Key-value	6.92	+0.83	+1.93
61.	↓60.	↑74.	Microsoft Azure Data Explorer	Relational, Multi-model	6.79	+0.20	+2.05
62.	↓61.	↑71.	Jackrabbit	Content	6.54	+0.09	+1.55
63.	↑66.	↑75.	Ignite	Multi-model	6.50	+0.48	+1.77
64.	↓63.	↓59.	Interbase	Relational	6.47	+0.34	+0.12
65.	↑68.	↓58.	Ingres	Relational	6.44	+0.67	-0.05
66.	↓62.	↓62.	Prometheus	Time Series	6.39	+0.12	+0.63
67.	↓65.	↓55.	Ehcache	Key-value	6.38	+0.29	-0.50
68.	↑69.	↓61.	SAP SQL Anywhere	Relational	5.76	+0.10	-0.19
69.	↑71.	↓66.	HyperSQL	Relational	5.74	+0.21	+0.56
70.	↑72.	↑73.	Microsoft Azure Table Storage	Wide column	5.64	+0.22	+0.70
71.	↓67.	↓63.	Aerospike	Key-value, Multi-model	5.61	-0.33	-0.06
72.	↓70.	↑79.	Graphite	Time Series	5.58	+0.00	+0.96
73.	↑78.	↓69.	ArangoDB	Multi-model	5.40	+0.67	+0.33
74.	↓73.	↑111.	Virtuoso	Multi-model	5.39	+0.02	+3.02
75.	↑76.	↓70.	Google Cloud Datastore	Document	5.25	+0.41	+0.23
76.	↓74.	↓64.	Derby	Relational	5.22	+0.25	-0.19
77.	↓75.	↑78.	SAP IQ	Relational	5.11	+0.26	+0.49
78.	↓77.	↓76.	Adabas	Multivalue	5.06	+0.32	+0.40
79.	↑80.	↓68.	OrientDB	Multi-model	5.03	+0.47	-0.10
80.	↑81.	80.	Oracle NoSQL	Multi-model	4.84	+0.43	+0.33
81.	↓79.	↓67.	OpenEdge	Relational	4.67	+0.01	-0.47
82.	82.	↑100.	TimescaleDB	Time Series, Multi-model	4.37	+0.15	+1.51
83.	83.	↓77.	MaxDB	Relational	4.23	+0.18	-0.41
84.	↑89.	↑85.	Google Cloud Bigtable	Wide column	4.17	+0.54	+0.46
85.	↑87.	↓84.	IBM Cloudant	Document	3.94	+0.18	+0.14
86.	86.	↓82.	Accumulo	Wide column	3.93	+0.05	-0.20
87.	↓84.	↓86.	SAP Advantage Database Server	Relational	3.92	-0.08	+0.33
88.	↑90.	↓81.	UniData,UniVerse	Multivalue	3.89	+0.27	-0.26
89.	↓88.	↑94.	RocksDB	Key-value	3.89	+0.19	+0.69
90.	↓85.	↑113.	ScyllaDB	Multi-model	3.88	-0.03	+1.73
91.	↑96.	↓89.	RavenDB	Document, Multi-model	3.82	+0.57	+0.47
92.	92.	↑110.	Google Cloud Spanner	Relational	3.69	+0.23	+1.22
93.	↓91.	↑96.	EXASOL	Relational	3.60	+0.07	+0.47
94.	↑95.	↑115.	TiDB	Relational, Multi-model	3.51	+0.25	+1.38
95.	↑99.	↓88.	PouchDB	Document	3.46	+0.48	+0.08
96.	↓93.	↑104.	Apache Druid	Multi-model	3.40	-0.04	+0.74
97.	↓94.	↓90.	Apache Phoenix	Relational	3.32	+0.06	+0.04
98.	↑102.	↓93.	InterSystems Caché	Multi-model	3.26	+0.36	+0.03
99.	↓98.	↓92.	LevelDB	Key-value	3.25	+0.17	+0.01
100.	↑101.	↓91.	Infinispan	Key-value	3.23	+0.29	-0.03
101.	↑107.	↓87.	Oracle Berkeley DB	Multi-model	3.11	+0.43	-0.29
102.	↓97.	↓83.	RethinkDB	Document, Multi-model	3.08	-0.06	-0.98
103.	↓100.	↑109.	4D	Relational	3.07	+0.11	+0.59
104.	↑105.	↓103.	Apache Drill	Multi-model	3.03	+0.30	+0.31
105.	↑109.	↓98.	IMS	Navigational	3.01	+0.38	+0.02
106.	↑108.	↑118.	Amazon Neptune	Multi-model	2.99	+0.36	+0.92
107.	↓103.	↑114.	GraphDB	Multi-model	2.93	+0.07	+0.79
108.	↓104.	↓95.	Apache Jena - TDB	RDF	2.90	+0.06	-0.26
109.	↓106.	↑166.	Trino	Relational	2.88	+0.19	+1.87
110.	110.	↓101.	Oracle Coherence	Key-value	2.67	+0.12	-0.15
111.	111.	↓102.	Percona Server for MySQL	Relational	2.54	+0.03	-0.21
112.	↑115.	↓99.	LMDB	Key-value	2.53	+0.29	-0.34
113.	113.	↓107.	CloudKit	Document	2.41	+0.12	-0.10
114.	↑118.	↓97.	RRDtool	Time Series	2.40	+0.32	-0.60
115.	↓112.	↓105.	JanusGraph	Graph	2.36	-0.03	-0.17
116.	↓114.	↑119.	EDB Postgres	Relational, Multi-model	2.36	+0.09	+0.31
117.	↑121.	↑163.	YugabyteDB	Relational, Multi-model	2.34	+0.39	+1.30
118.	↓116.	↓108.	Amazon CloudSearch	Search engine	2.31	+0.10	-0.18
119.	↑120.	↑147.	TigerGraph	Graph	2.24	+0.22	+0.91
120.	↓117.	↓112.	Amazon SimpleDB	Key-value	2.18	0.00	-0.03
121.	↓119.	↑124.	Tibero	Relational	2.04	+0.01	+0.16
122.	↑124.	↑137.	Stardog	Multi-model	1.98	+0.09	+0.52
123.	↓122.		SpatiaLite	Spatial DBMS, Multi-model	1.98	+0.04	
124.	↓123.	↑125.	IBM Db2 warehouse	Relational	1.94	+0.03	+0.07
125.	↑138.	↓117.	GridGain	Multi-model	1.94	+0.39	-0.14
126.	↑136.	↓122.	jBASE	Multivalue	1.92	+0.30	-0.05
127.	↓125.	↓120.	MonetDB	Relational, Multi-model	1.86	-0.02	-0.18

https://db-engines.com/en/ranking

Materials

- Moodle will be adopted for:
  - Course information
  - Lecture and lab materials
  - Communication and discussions

- Contact: office I229 / InfoLab (I123) / by email

Next steps

- Answer the 'BDNR Survey' (if you haven't done so)
- Prepare for the next lecture:
  - organize groups before class (register in Moodle, you can change this later)
  - explore NoSQL technologies to propose topic
- Prepare personal setup
  - Git
  - Docker

What are your expectations?

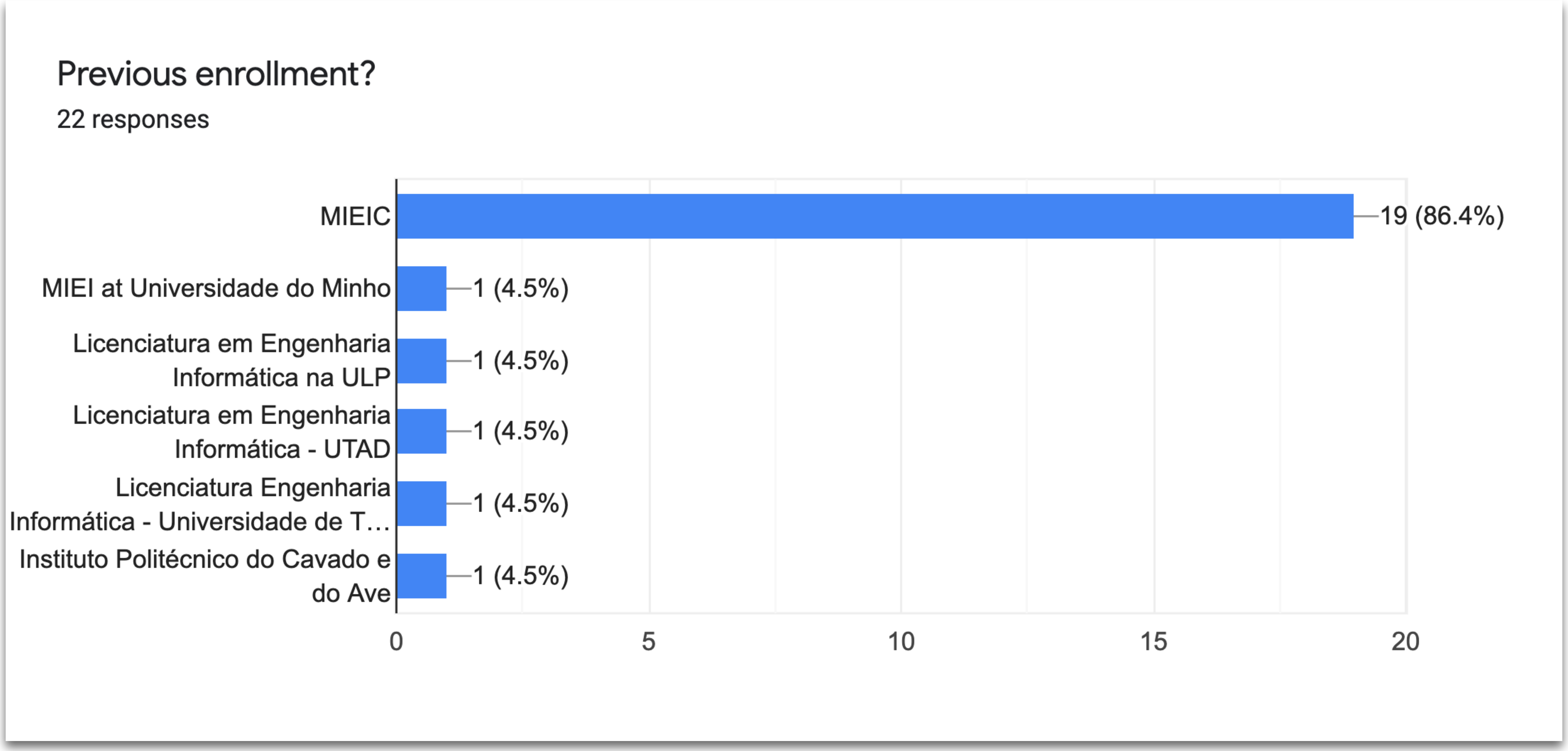
References

- NoSQL Distilled, Pramod J. Sadalage and Martin Fowler. Addison-Wesley, 2012
- Next Generation Databases, Guy Harrison. Apress, 2016

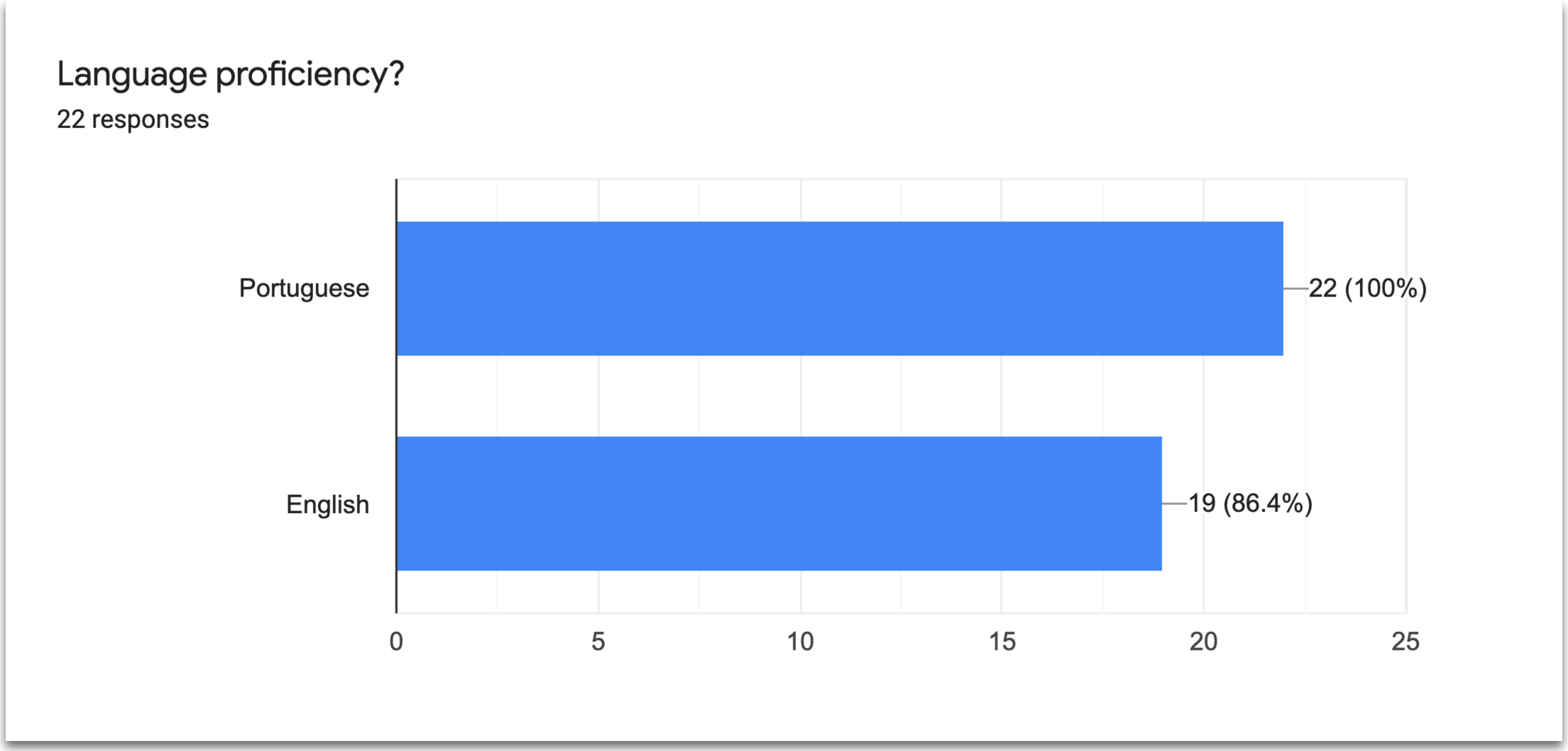
- Bigtable: A Distributed Storage System for Structured Data (2006) https://static.googleusercontent.com/media/research.google.com/en//archive/bigtable-osdi06.pdf
- Dynamo: Amazon's Highly Available Key-value Store (2007) https://www.allthingsdistributed.com/files/amazon-dynamo-sosp2007.pdf

Student Survey

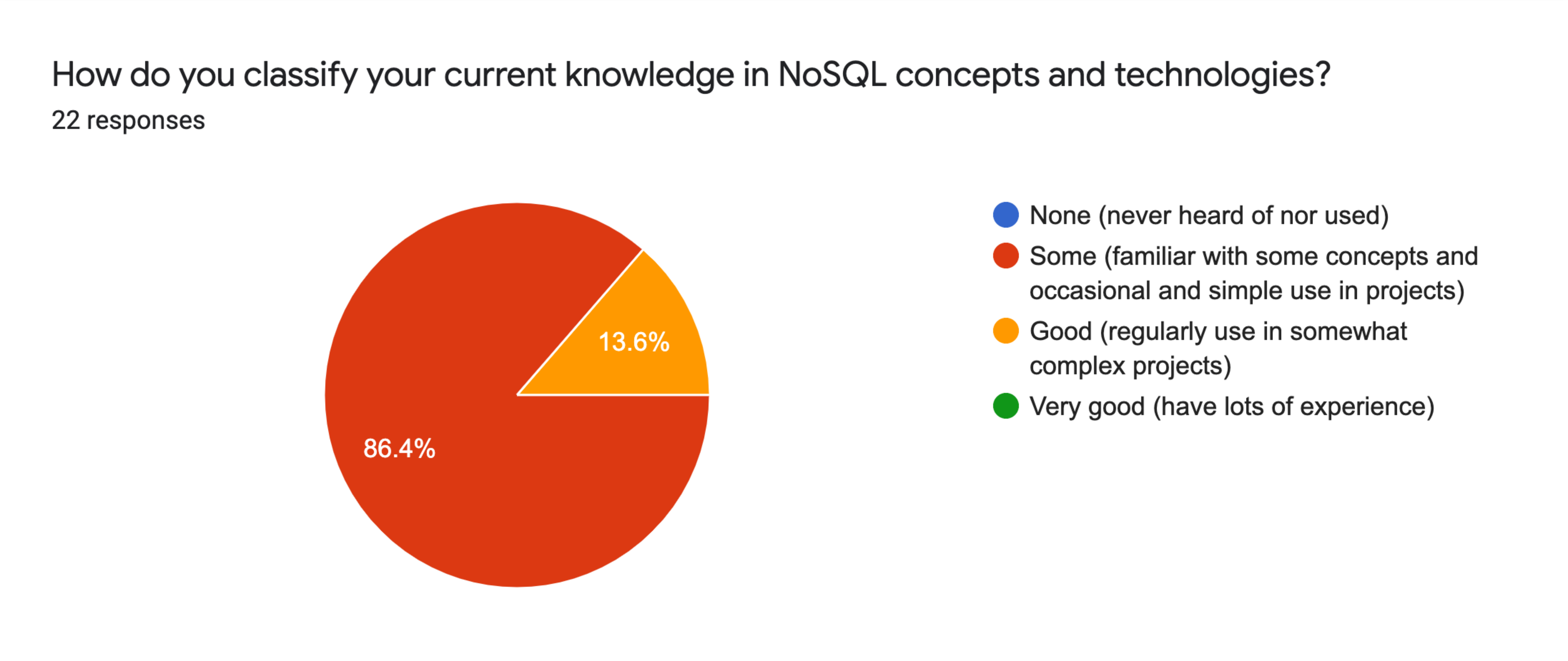
Previous Enrollment



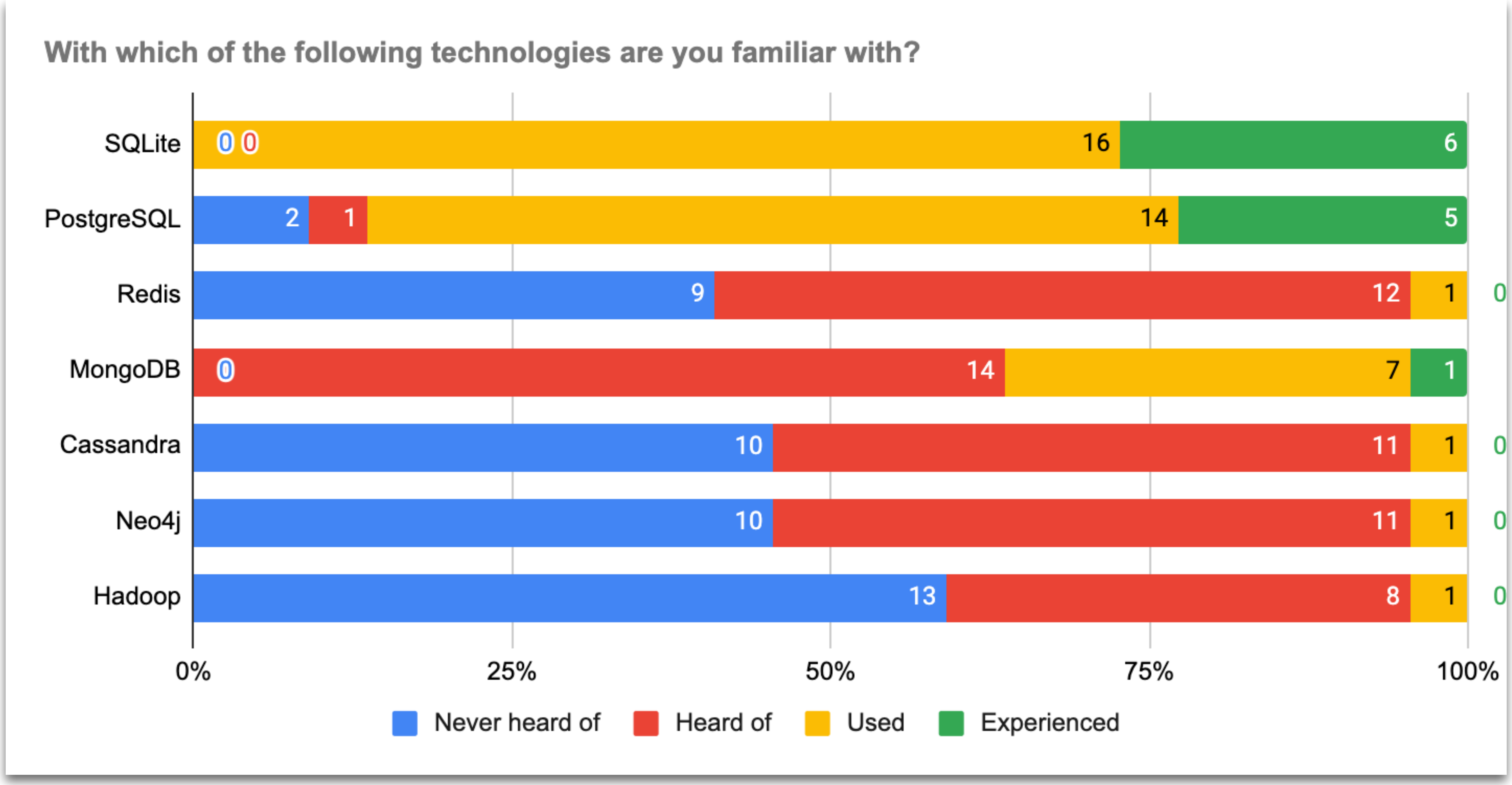
Language Proficiency



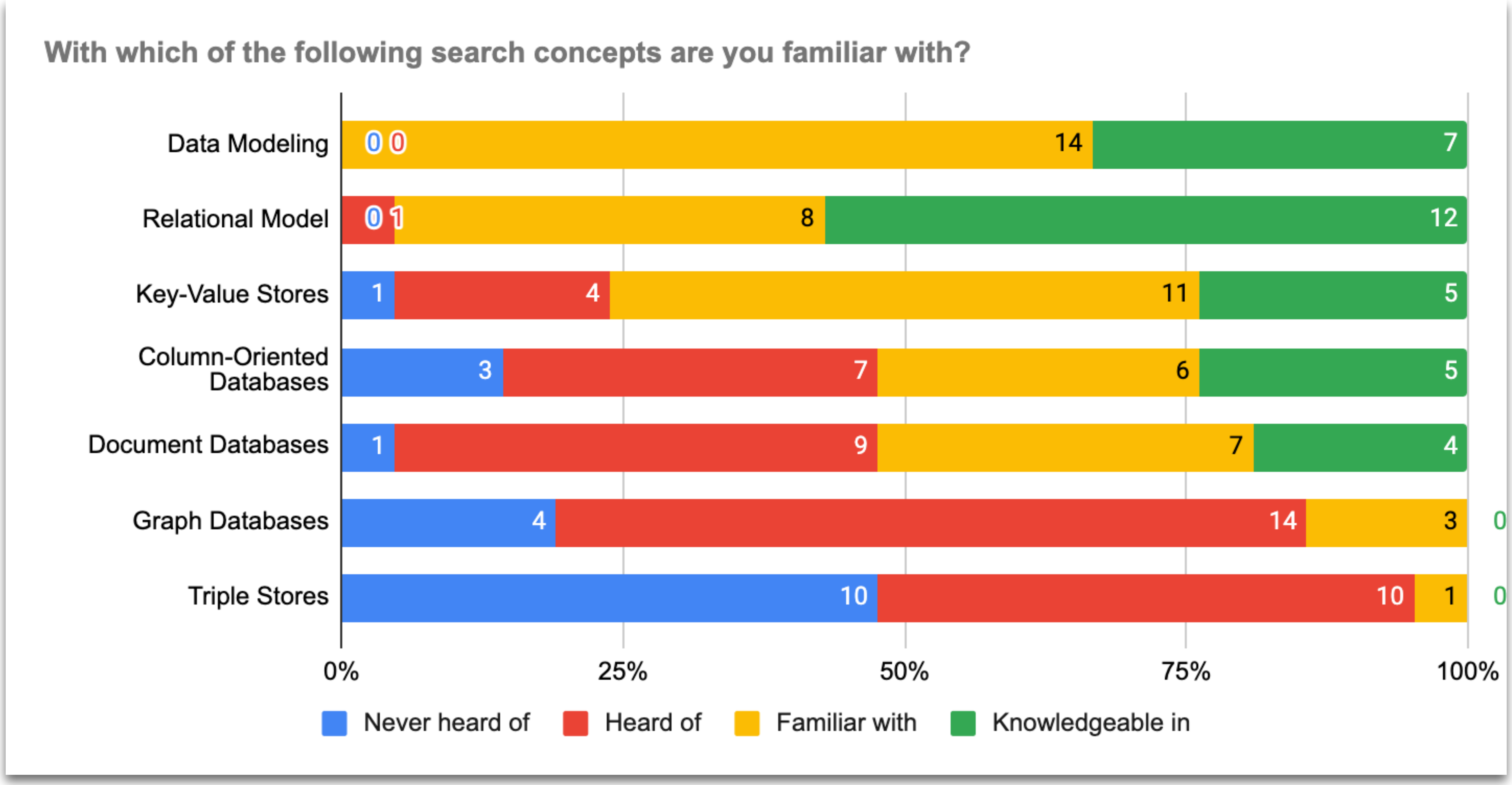
NoSQL Experience



Experience with Technologies



Familiar with Concepts



Overall Interests